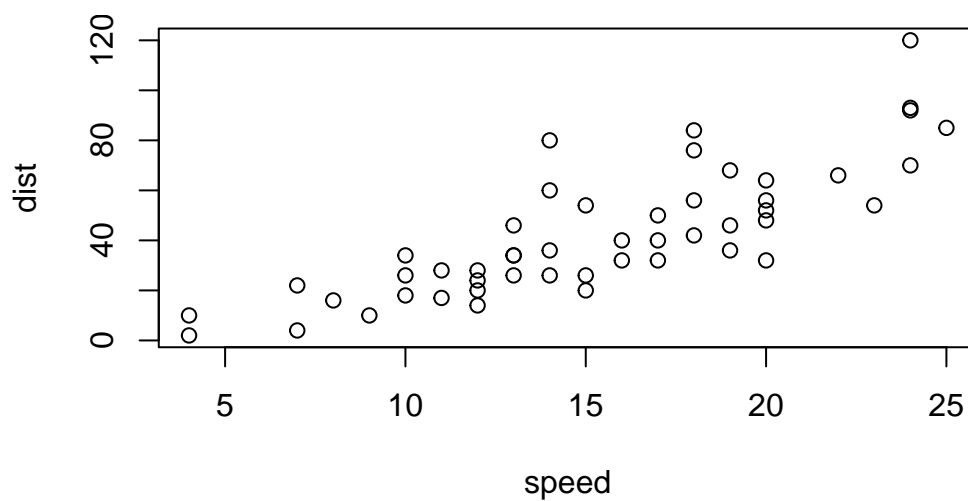# Class 5: Data visualization with ggplot2

Jie

#Our first plot

R has base graphics

```
plot(cars)
```



```
cars
```

```
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
7    10   18
```

|		
8	10	26
9	10	34
10	11	17
11	11	28
12	12	14
13	12	20
14	12	24
15	12	28
16	13	26
17	13	34
18	13	34
19	13	46
20	14	26
21	14	36
22	14	60
23	14	80
24	15	20
25	15	26
26	15	54
27	16	32
28	16	40
29	17	32
30	17	40
31	17	50
32	18	42
33	18	56
34	18	76
35	18	84
36	19	36
37	19	46
38	19	68
39	20	32
40	20	48
41	20	52
42	20	56
43	20	64
44	22	66
45	23	54
46	24	70
47	24	92
48	24	93
49	24	120
50	25	85

```r
head(cars)
```

```
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
```
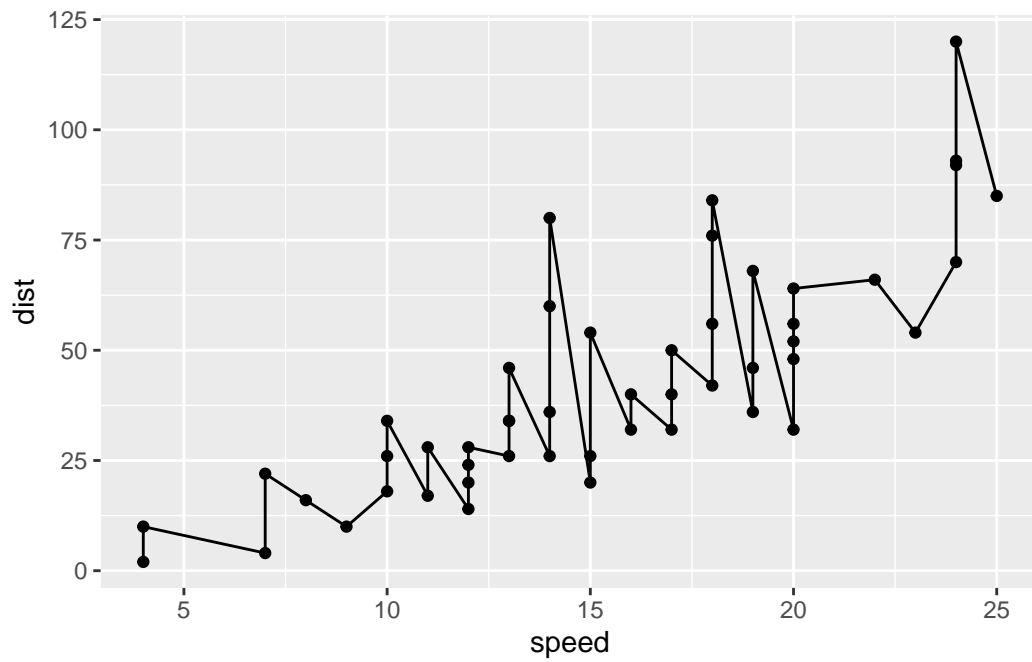
```r
library(ggplot2)
```

3 layers of ggplot - *Data* - *Aes* - *Geoms*

```r
ggplot(data=cars) + aes(x=speed,y=dist) + geom_point()
```
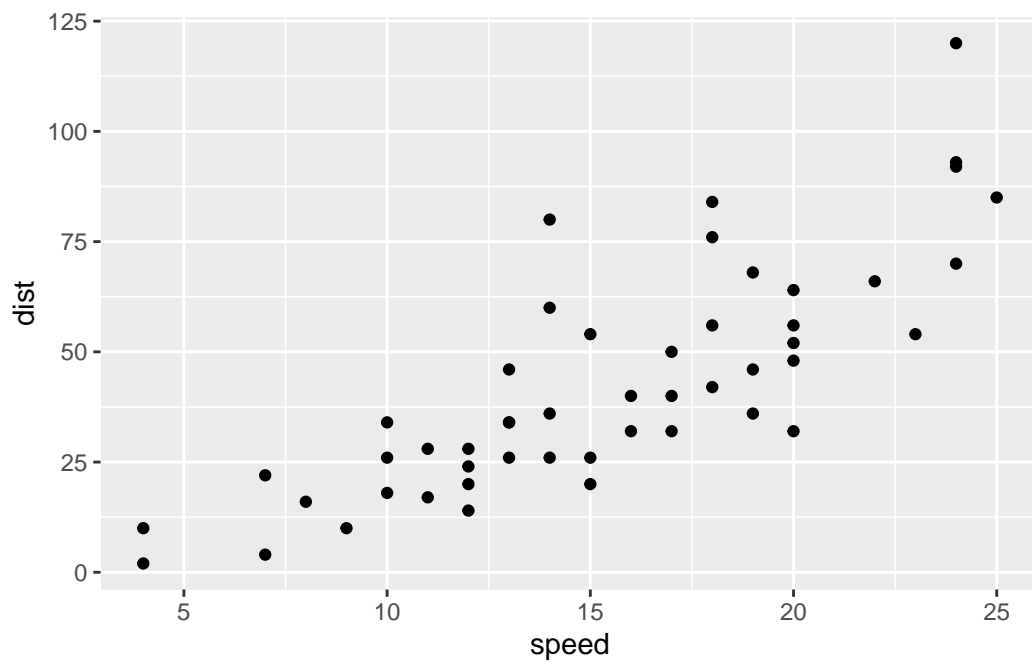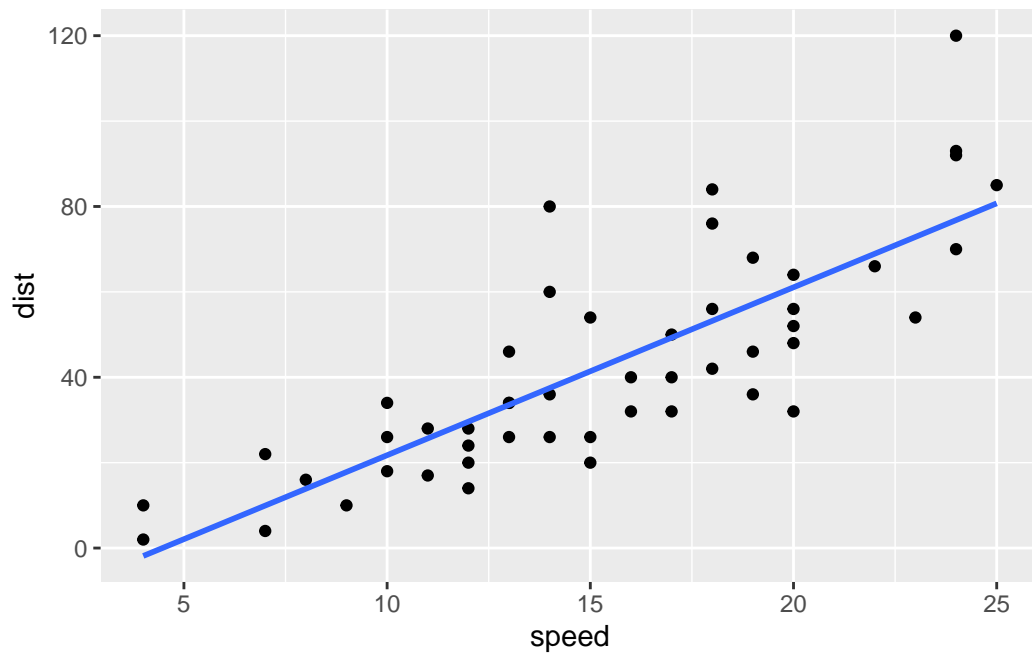


```r
ggplot(data=cars) + aes(x=speed,y=dist) + geom_point() + geom_line()
```

```
ggplot(data=cars) + aes(x=speed,y=dist) + geom_point() + geom_smooth(method=lm, se=FALSE)
```

`geom_smooth()` using formula 'y ~ x'

4

```
ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  labs(title="Speed and Stopping Distances of Cars",
       x="Speed (MPH)",
       y="Stopping Distance (ft)",
       subtitle = "Your informative subtitle text here",
       caption="Dataset: 'cars'") +
  geom_smooth(method="lm", se=FALSE) +
  theme_bw()
```

`geom_smooth()` using formula 'y ~ x'

Speed and Stopping Distances of Cars
Your informative subtitle text here

Dataset: 'cars'
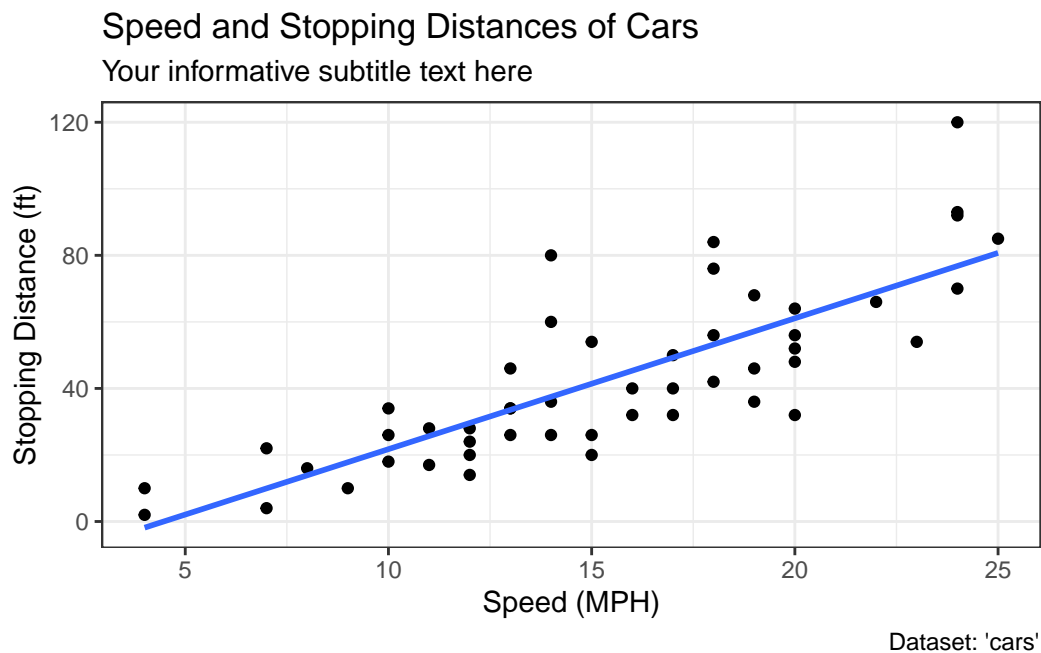
```
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)
head(genes)
```

```
      Gene Condition1 Condition2       State
1     A4GNT -3.6808610 -3.4401355 unchanging
2      AAAS  4.5479580  4.3864126 unchanging
```

```
3         AASDH  3.7190695  3.4787276 unchanging
4          AATF  5.0784720  5.0151916 unchanging
5          AATK  0.4711421  0.5598642 unchanging
6 AB015752.4 -3.6808610 -3.5921390 unchanging
```
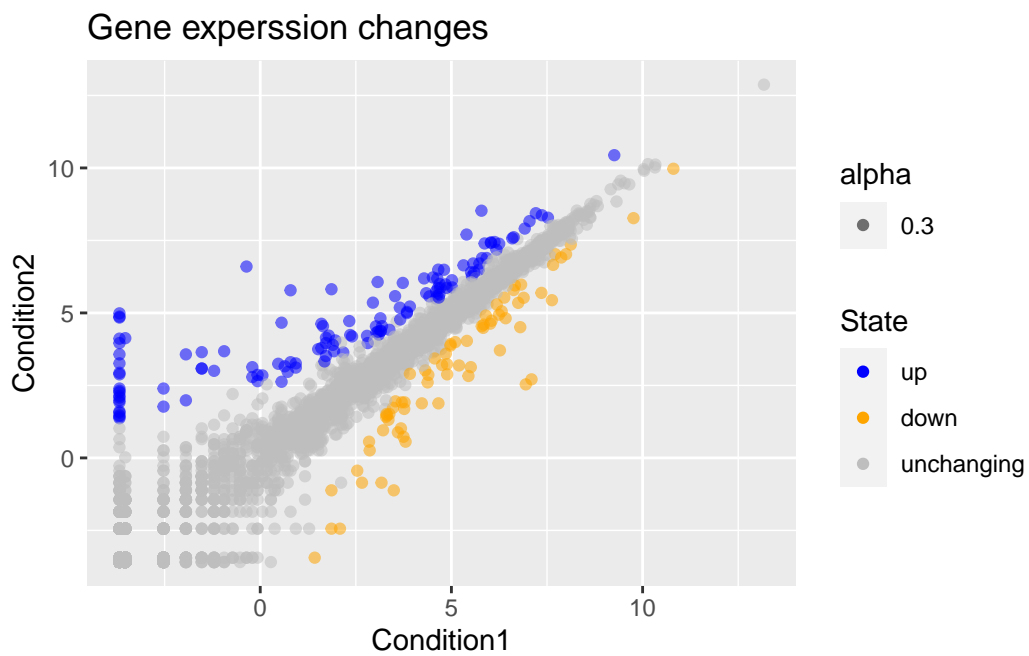
```r
gene_graph <- ggplot(genes) +
    aes(x=Condition1, y=Condition2) +
    geom_point(color = "blue")
gene_graph
```
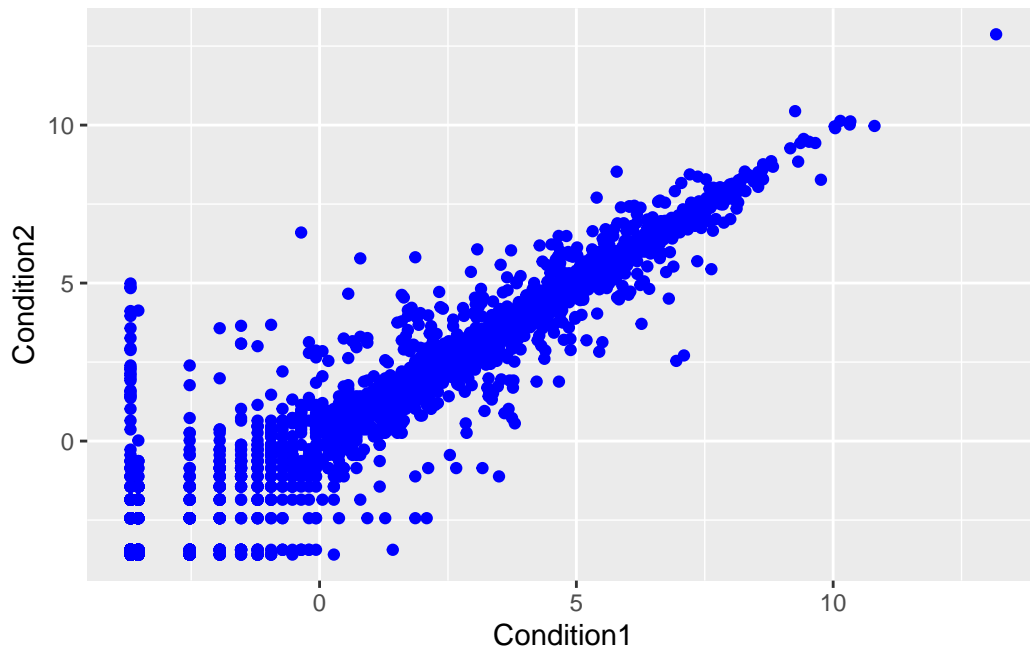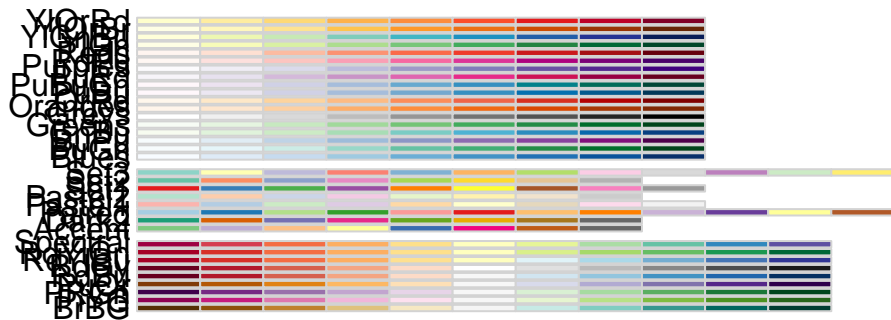


```r
gene_graph <- ggplot(genes) +
    aes(x=Condition1, y=Condition2, col=State,alpha=0.3) +
    geom_point()+labs(title = "Gene experssion changes")
gene_graph
```

Gene experssion changes
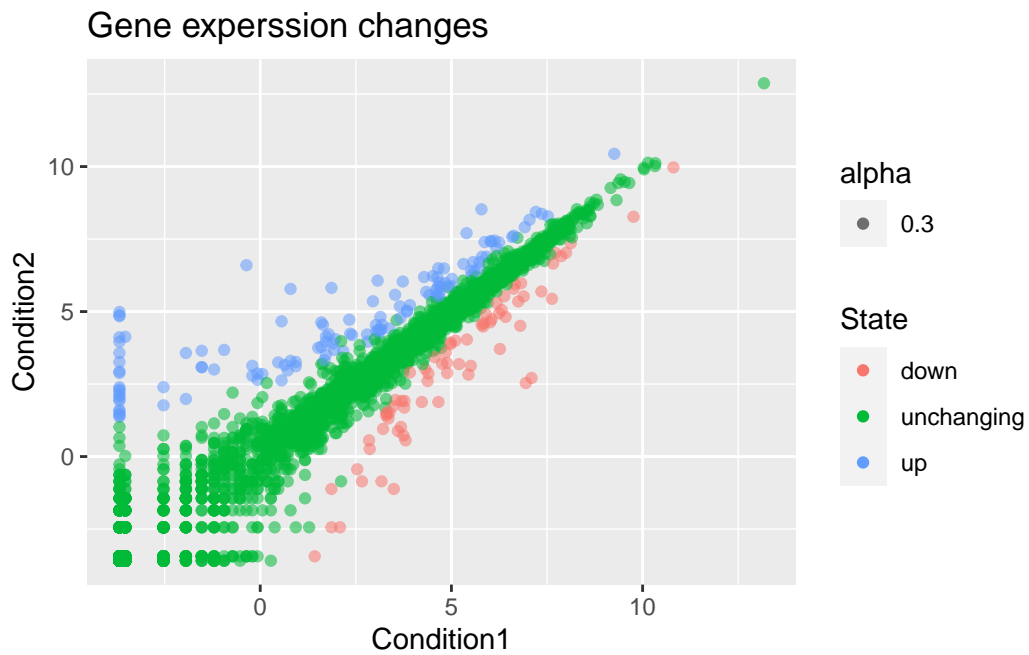
```
gene_graph + scale_color_manual(values = c("up" = "blue","down"="orange","unchanging"="gre
```


Gene experssion changes

```
library(RColorBrewer)
display.brewer.all()
```



```
gene_graph <- ggplot(genes) +
    aes(x=Condition1, y=Condition2, col=State) +
    geom_point()+labs(title = "Gene experssion changes", x="Control (no drug) ",
        y="Drug Treatment")
gene_graph + scale_color_manual(values = brewer.pal(3,"Pastel1") )
```

```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.
gapminder <- read.delim(url)

library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':
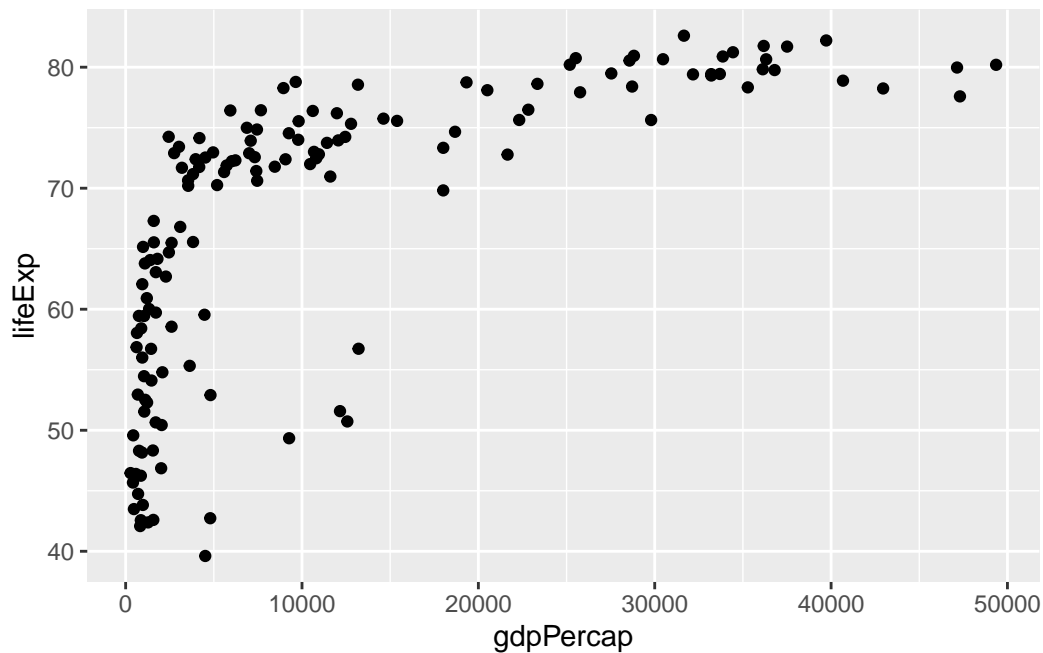
    filter, lag

The following objects are masked from 'package:base':
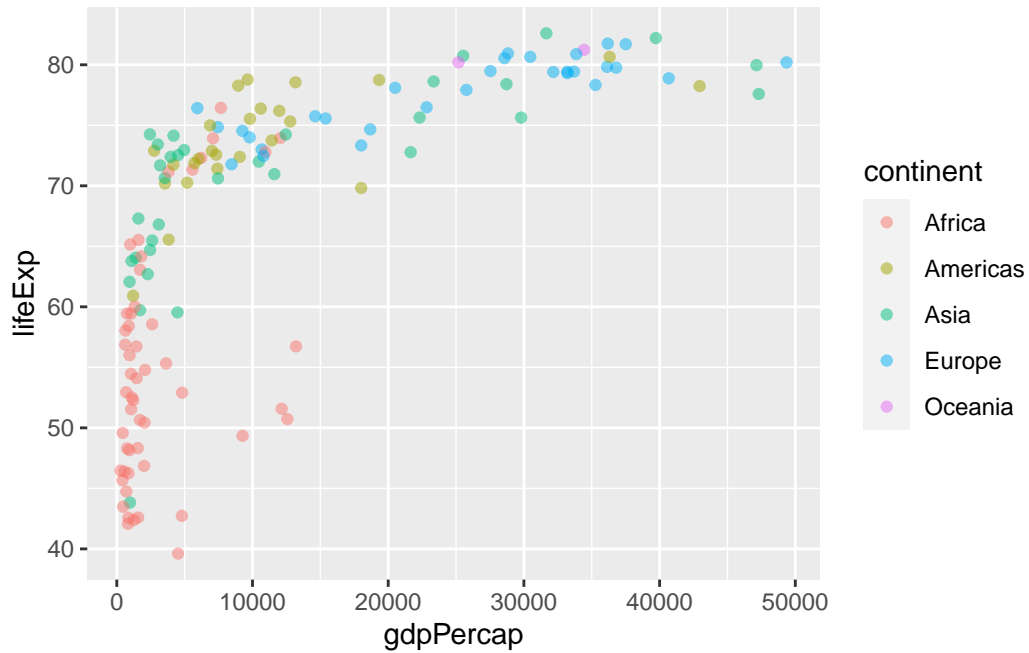
    intersect, setdiff, setequal, union

```
gapminder_2007 <- gapminder %>% filter(year==2007)
```
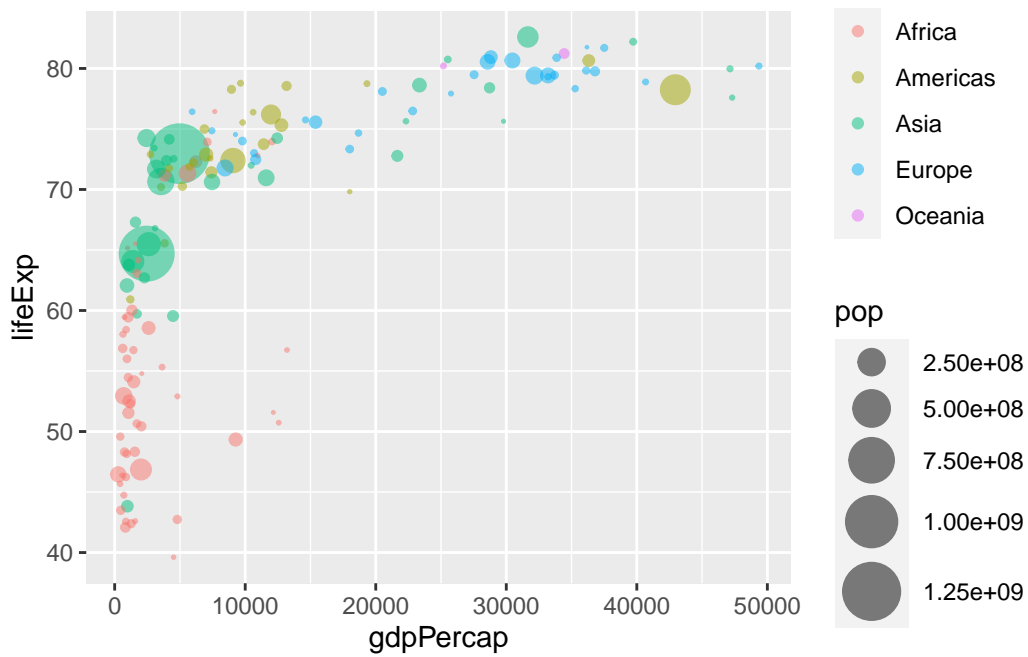
```
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point()
```
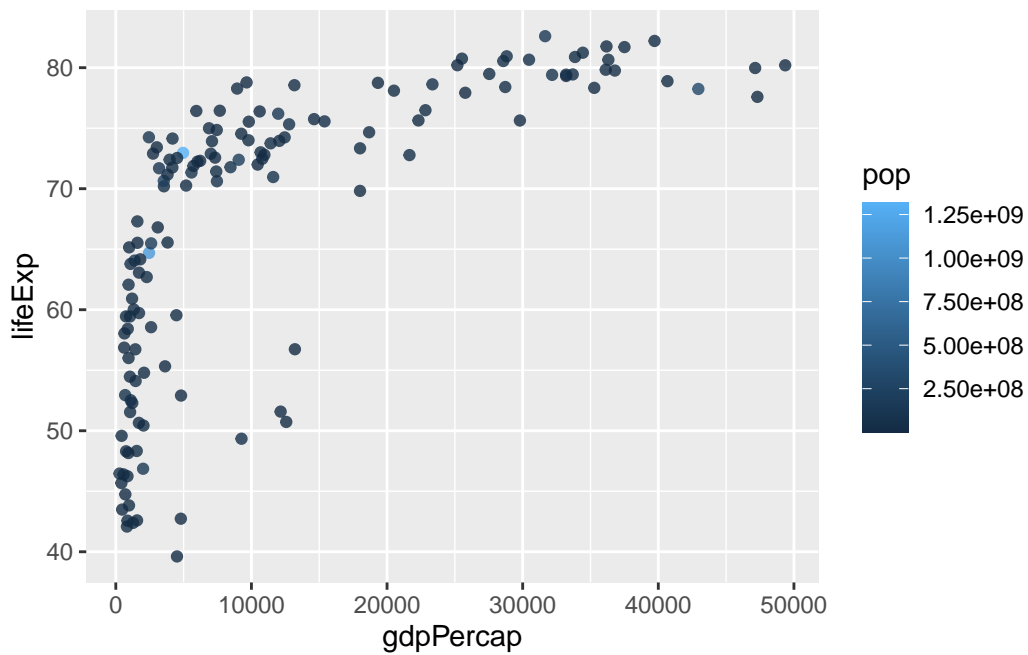
```
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp,color=continent) +
  geom_point(alpha=0.5)
```
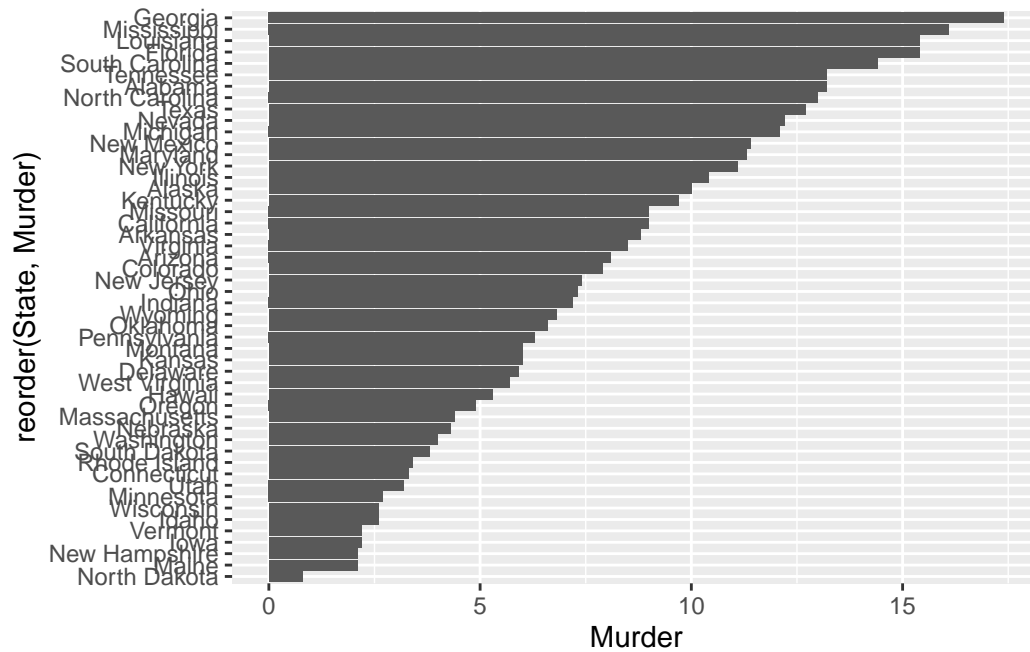


```
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp,color=continent,size=pop) +
  geom_point(alpha=0.5) +
  scale_size_area(max_size = 10)
```

```
ggplot(gapminder_2007) +
  aes(x = gdpPercap, y = lifeExp, color = pop) +
  geom_point(alpha=0.8)
```

```
USArrests$State <- rownames(USArrests)
ggplot(USArrests) +
  aes(x=reorder(State,Murder), y=Murder) +
  geom_col() +
  coord_flip()
```


barplot?